

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

BAKALÁŘSKÁ PRÁCE

2010

Filip Bartmann

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Absolvování individuální odborné praxe
Individual Professional Practise in the
company

„Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

Na úvod zprávy o bakalářské praxi bych chtěl poděkovat Ing. Janu Guiziurovi za možnost spolupráce s firmou Testudo, za poskytování potřebné součinnosti v průběhu řešení svěřených úkolů a za důvěru, se kterou mi je svěřil.

Abstrakt

Cílem práce bylo vytvořit pro potřeby firmy Testudo corp. s.r.o. systém, který bude sloužit k realizaci plateb od klientů, ovládání webhostingu, pro evidenci klientů a bude komunikovat se systémem Raifaisen Bank (e-platby).

Pro řešení je použit software, který má zadavatel k dispozici. CMS Joomla! 1.5, systém WHMCS a ekonomický a informační systém POHODA 2010. Bylo navrženo a realizováno propojení systémů. K propojení byly využity existující rozhraní a nástroje. Integrátor J!WHMCS, API Raifaisen Bank a rozhraní ekonomického systému, jejichž nastavení byla provedena podle požadavků jednotlivých systémů.

Pro dosažení požadované funkčnosti byly doprogramovány komponenty pro CMS Joomla!. Vznikla komponenta pro zpracování importů do ekonomického a informačního systému POHODA 2010 a komponenta pro správu a nastavení účtů jak pro administrátora tak pro klienta.

Abstract

The purpose of the work was to create for needs of company Testudo corp. s.r.o. the system which will serve for realization of client's payments, webhosting control, for client's evidence and will communicate with Raifaisen Bank system (e-payments).

For solving the software which submitter has to disposal, CMS Joomla! 1.5, system WHMCS and economic and informative system POHODA 2010, was used.

Connection of systems was projected and realized. For connection the existing interface and instruments were exploited. Integrator J!WHMCS, API Raifaisen Bank and economic system' interface, whose setting were made acc to requirements of separate systems.

For reaching of inquired utility the components for CMS Joomla! were additionally programmed. Component for execution of imports into economic and informative system POHODA 2010 and component for administration and account setting for administrator as well as for client originated.

Klíčová slova

CMS Joomla!, WHMCS, J!WHMCS, POHODA, uživatel, klient, platba, faktura

Keywords

CMS Joomla!, WHMCS, J!WHMCS, POHODA, user, client, payment, invoice

Seznam použitých symbolů a zkratek

API – application programmer interface – aplikační programátorské rozhraní

CMS – Content Management System – systém pro správu obsahu

FTP – File Transfer Protocol – protokol pro přenos souborů

Joomla – CMS Joomla!

J!WHMCS – Joomla Web Hosting Management Complete Solution – bridge propojující CMS Joomla! a WHMCS

MVC – Model – View – Controller – návrhový vzor pro tvorbu aplikací oddělující kód od dat

POHODA – ekonomický a informační systém POHODA 2010

RPC – Remote Procedure Call – protokol pro vzdálenou komunikaci

RSS – skupina formátů pro odběr novinek z www stránek

SQL – Structured Query Language – jazyk pro práci s databázema

URL – Uniform Resource Locator - slouží k přesné specifikaci umístění zdrojů informací

VS – variabilní symbol

WHMCS – Web Hosting Management Complete Solution – systém pro platební podporu a podporu(helpdesk)

XML – Extensible Markup Language – tagovací značkovací jazyk

csv – comma separated value – hodnoty oddělené středníkem

.htaccess – soubor umožňující běžným uživatelům upravovat nastavení webového serveru (nejčastěji Apache)

http – hypertext transfer protocol – protokol pro přenos hypertextu

#__ – variabilní prefix tabulek v Joomla

Obsah

1 Úvod.....	2
1.1 Popis firmy.....	2
1.2 Popis systému CMS Joomla!.....	2
1.3 Popis pozice studenta.....	2
2 Seznam zadaných úkolů.....	2
3 Postup řešení zadaných úkolů.....	3
3.1 Propojení CMS Joomla! s WHMCS.....	3
3.1.1 Popis systému WHMCS.....	3
3.1.2 Propojení systémů.....	3
3.2 Ověření plateb do RB.....	4
3.2.1 Realizace propojení systému RB (e-platby) s WHMCS.....	4
3.2.2 Ověřování plateb	5
3.3 Propojení CMS Joomla! s ekonomickým a informačním systémem POHODA 2010.....	6
3.4 Propojení systému s registrací domén u domena.cz.....	7
3.5 Komponenta do CMS Joomla! na správu virtuálních serverů.....	7
4 Znalosti ze studia.....	11
5 Scházející znalosti.....	11
6 Dosažené výsledky.....	11
Použitá literatura.....	13

1 Úvod

1.1 Popis firmy

Firma Testudo má za cíl propagaci a rozšiřování CMS Joomla! mezi českými a slovenskými uživateli. K tomuto účelu provozuje portály www.joomlaportal.cz a pro podporu e-shopů postavených na CMS Joomla! www.obchod-virtuemart.cz. Cílem těchto portálů je poskytovat podporu začínajícím i pokročilým uživatelům, tvořit národní lokalizaci a podílet se na vývoji CMS Joomla! a jejích doplňků.

Komě těchto neziskových projektů poskytuje společnost komerční služby spojené s CMS Joomla! - programování a tvorbu webových stránek a e-shopů na klíč, konzultační služby, provozuje internetový obchod a webhosting.

1.2 Popis systému CMS Joomla!

CMS Joomla! Je bezplatný publikační systém. Je napsán v jazyce PHP5 a jako databázi používá MySQL. Jedná se o komplexní systém obsahující potřebné nástroje jako zobrazení článků ve formě blogů, vlastní šablonovací systém, ankety nebo práci s RSS. Nástroje pro správu komunity, diskusní fórum, fotogalerii nebo e-shop, lze stáhnout jako doplněk. Tyto doplňky se dělí na komponenty, moduly a pluginy.

Vznikl jako odnož systému Mambo 4.5.2.3. 17. srpna 2005 po neshodách skupiny vývojářů s mateřskou firmou Miro. Samotné jméno Joomla bylo oznámeno 1. září 2005. Verze 1.0 systému, která zahrnovala pouze upravenou administrační šablonu a opravené bezpečnostní chyby vyšla 16. září.

Verze 1.5, která byla téměř kompletně přepracována podle standardu MVC a PHP 5 byla vydána 22. ledna 2008. V současné době je vývoj zaměřen na verzi 1.6, která přinese revoluční novinky ohledně přístupových práv.

Komponenta je hlavní typ rozšíření systému. Přidává do aplikace rozšíření funkčnosti. Modul slouží k zobrazení informací na určeném místě v šabloně. Poslední typ rozšíření v předchozích verzích nazývaný Mambot, nyní Plugin, se dělí na několik typů:

- Obsahový – umožňuje manipulaci s obsahem
- Vyhledávací – rozšiřuje základní vyhledávání
- Autentifikační – stará se o přihlašování uživatelů
- Storage uživatelů
- Editory – obsahuje WYSIWYG editory pro jednoduchou tvorbu obsahu
- Systémové – umožňují pracovat se systémovými funkcemi Joomla a upravovat výstup z jádra Joomla
- XML-RPC – umožňují vzdálený přístup k funkcím Joomla

1.3 Popis pozice studenta

Spolupráci s firmou Testudou corp. a s jejím ředitelem, Ing. Janem Guiziurem jsem navázal v roce 2007, jako pokročilý uživatel systému CMS Joomla!. Tato spolupráce se s postupem času prohlubovala a dnes si dovoluji říci, že patřím k jejím externím spolupracovníkům na pozici programátor. Na základě navázané spolupráce mi byly svěřeny k řešení úkoly, které tvoří náplň mé odborné praxe.

2 Seznam zadaných úkolů

1. Propojení CMS Joomla! s WHMCS – pracnost řešení 5 dnů
2. Ověření plateb do Raifaisen Bank (e-platby) a propojení se systémem WHMCS – pracnost řešení 15 dnů

3. Propojení systému s ekonomickým a informačním systémem POHODA 2010 (import faktur a klientů) – pracnost řešení 15 dnů
4. Propojení systému s registrací domén u domena.cz – pracnost řešení 2 dny
5. Komponenta do CMS Joomla! na správu virtuálních serverů – pracnost řešení 13 dnů

3 Postup řešení zadaných úkolů

3.1 Propojení CMS Joomla! s WHMCS

Mým prvním úkolem bylo propojení CMS Joomla se systémem pro správu plateb, klientů a pro podporu online obchodování WHMCS.

3.1.1 Popis systému WHMCS

WHMCS je platební systém umožňující platby v různých měnách, jejich správu a automatizaci.

Jeho součástí je systém pro podporu klientů a HelpDesk. HelpDesk je řešený pomocí ticketů s nabídkami možných řešení na právě pokládanou otázku. Pro časté dotazy nabízí bázi znalostí.

Dále obsahuje nástroje pro práci s doménami včetně možnosti jejich registrace nebo transferu.

S tímto složitým systémem jsem se seznámil pouze v rozsahu potřebném pro splnění zadaných úkolů.

3.1.2 Propojení systémů

Propojení systémů je zajištěno existujícím bridgem systému WHMCS nazvaným J!WHMCS integrátor pro Joomla!.

Bridge se skládá z komponenty, která zajišťuje komunikaci systémů, ze dvou pluginů pro autentifikaci a ze systémových pluginů pro propojení jazykových systémů. Tato část bridge je rozšířením systému CMS Joomla!. Dále bridge obsahuje adresáře a soubory, které se kopírují do WHMCS. V těchto souborech jsou mimo jiné dvě speciální šablony s kódem zajišťujícím propojení systémů na úrovni zobrazování. Každý systém má jinak řešen šablonovací systém. Aby se html stránky korektně zobrazovaly, je potřebný zmiňovaný kód.

Po provedení instalačních prací následovalo nastavení integrátoru. V úloze k tomu určené se zadává licenční klíč a hodnoty globálních parametrů, rozčleněných do bloků pro Joomla, WHMCS a bloky, ve kterých dochází k párování položek menu na www stránkách Joomla se stránkami WHMCS, aby k aktivní položce menu v Joomla bylo možno správně přiřadit parametr id s hodnotou active_menu.

Nastavení bloku pro Joomla spočívá v zadání URL k Joomla a k obrázkům v Joomla, URL stránek Joomla pro přihlášení a odhlášení, pro přepis http na https na vygenerované stránce. V tomto bloku se také nastavuje, zda lze testování nastavení spouštět mimo systém.

V bloku parametrů pro WHMCS se nastavuje uživatelské jméno a heslo pro komunikaci mezi API systémů a URL WHMCS.

V autentizačním zásuvném modulu Joomla se nastavuje způsob synchronizace uživatelů mezi tabulkami uživatelů, které má každý systém vlastní.

Bridge má vestavěnou funkci pro otestování správně zadaného licenčního klíče, což je první základní podmínkou pro ověření další funkčnosti, a funkci, která kontroluje nastavení.

Při prvotním nastavování, které jsem prováděl na v té chvíli dostupné nižší verzi systému, která nedisponovala současným GUI, se vyskytly problémy, které byly posléze detekovány jako nezapisovatelnost do některých adresářů. Problémy byly vyřešeny ve spolupráci s podporou dodavatele

systému.

Další řešený problém vznikl při vývoji pluginu pro registraci domén u domena.cz (provozovatel Ignum). Systém hlásil nevalidní licenci. Tento problém byl nejprve neúspěšně řešen s podporou dodavatele J!WHMCS. Na základě skutečnosti, že J!WHMCS nemá zdrojový kód šifrovaný pomocí IONCUBE jako WHMCS, proto jsem z něj mohl vyčíst, že příčinou je nastavení příliš jemného hlášení chyb PHP v .htaccess.

WHMCS i Bridge jsou často aktualizovány. Několik aktualizací prováděl i v průběhu vývoje. Aktualizace WHMCS spočívá v přepsání souborů na FTP serveru a aktualizace bridge v reinstalacích balíčků. Nastavení je při instalaci zachováno.

3.2 Ověření plateb do RB

3.2.1 Realizace propojení systému RB (e-platby) s WHMCS

Předmětem úkolu je řešení podpory procesů, vznikajících při objednávce zboží v e-shopu WHMCS a platbě klienty, prostřednictvím tohoto systému.

Klient pracuje ve firemním e-shopu, který je součástí WHMCS. Formulář, se kterým pracuje, je aplikováním propojení šablon systému Joomla a WHMCS, jak je popsáno v kapitole 3.1.2. Po provedení objednání z firemní nabídky systém zobrazí seznam veškerých objednaných služeb, které se váží k probíhající objednávce s možností úpravy. Přihlášenému klientovi systém nabídne předvyplněné fakturační údaje. Klient, který nemá údaje vedeny v databázi WHMCS, nebo není přihlášen, musí tyto údaje vyplnit. Při dokončení objednávky proběhne zápis údajů do WHMCS.

V dalším kroku se ze systému WHMCS klientovi zobrazí faktura s možností zadat její úhradu. Po potvrzení této činnosti probíhá přesměrování z WHMCS do systému RB.

Systém WHMCS má pro implementaci neobsažených platebních bran pro uživatele, kteří si jej zakoupili, k dispozici předpřipravený kód, který obsahuje veřejné rozhraní systému, do kterého je zapotřebí doplnit kód podle API dané banky, kde má uživatel účet. Toto je řešeno jednou z funkcí scriptu php nazvaného platební brány. Pro každou platbu je bance generován z WHMCS metodou GET, to je v URL, formulář s údaji. Součástí URL jsou bankou specifikované parametry. Ze všech nabízených parametrů jsem použil:

- identifikační řetězec prodejce, sloužící k přiřazení platby prodejci
- kód banky
- částku k proplacení
- kód měny převodu
- variabilní symbol platby, který přiřazuje platbě systém WHMCS
- adresu stránky, která se zobrazí po úspěšném zadání platebního příkazu
- adresu stránky, která se zobrazí po neúspěšném zadání platebního příkazu

Na stránkách banky klient podle požadavků, specifikovaných pro testovací prostředí, vyplňuje klientské číslo a autentizační kód. Otevře se formulář pro platbu za zboží a služby pro testovací prostředí, na kterých zadá vše potřebné pro platbu a platbu potvrdí. V této chvíli je řešen klientský požadavek v bance.

Ve chvíli, kdy je požadavek vyřešen, banka na základě údajů, které s platbou obdržela, zasílá informace o tom, jak byl požadavek vyřešen. Podle toho, zda byl vyřešen úspěšně či neúspěšně, se liší odpověď. Adresu stránek jsem specifikoval pro případ úspěšného i neúspěšného zadání platebního příkazu. Bankovní systém k adresám stránek, jež slouží jako návrat na stránky prodejce, přidává hodnotu vstupního parametru variabilní symbol platby. Adresace směřuje do systému Joomla.

Komunikaci s bankou jsem testoval na testovacím rozhraní banky, které obsahuje 3 fiktivní účty, z čehož jeden je pro testovací úkoly bez prostředků.

3.2.2 Ověřování plateb

Požadavek byl řešen přes další rozšíření CMS Joomla! po předchozí realizaci propojení obou systémů.

Pro možnost komunikovat s více bankami jsem vytvořil v systému Joomla pluginovatelnou komponentu whmcspay, nejedná se tedy o rozšíření typu plugin, ale o komponentu, kterou jsem vytvořil vlastními pluginy.

V MySQL v systému CMS Joomla! jsem založil pomocnou tabulku #__whmcspay_payments (Tab. 1), propojenou s daty WHMCS přes jejich vlastní klíč, jehož obsahem je variabilní symbol.

Sloupec	Typ	Nulový	Výchozí	Komentáře
id	int(5)	Ne		Primární klíč Joomla
plugin_id	int(5)	Ne	0	ID pluginu pro platební bránu
checked_out	int(5)	Ne	0	Interní zamykací pole Joomla s ID uživatele editujícího položku
checked_out_time	datetime	Ne	0000-00-00 00:00:00	Čas editace uživatelem
payed	int(1)	Ne	0	Příznak zaplacení
var_symbol	varchar(255)	Ne		Variabilní symbol
proforma	int(1)	Ne	0	Příznak, zda byla importována proforma faktura
final	int(1)	Ne	0	Příznak, zda byla importována faktura
gateway	varchar(255)	Ne		Název platební brány

Tab. 1: tabulka #__whmcspay_payments

Po dokončení příkazu k platbě klientem operací ve WHMCS přes webové rozhraní nechávám provést přesměrování údajů o platbě podle variabilního symbolu do Joomla, jak uvádím v předchozí podkapitole. Z Joomla volám metodou PHP URL banky, na které se dynamicky vytváří přehled transakcí prodejce, s použitím parametrů:

- identifikační řetězec prodejce
- heslo
- číslo účtu prodejce
- kód banky
- variabilní symbol
- formát stránky s volbou PLAIN (csv)
- Z hodnot řádků na HTML stránce zpracovávám údaje:
- poslední den splatnosti
- stav příkazu

Po vyhodnocení stavu příkazu pro získání hodnoty položky payed zakládám pro hodnotu variabilního symbolu do tabulky #__whmcspay_payments záznam.

Pokud je stav příkazu 2 – Zrealizováno, přiřazuji pro payed hodnotu 1, volám WHMCS a přes jeho definované rozhraní zadávám pro záznam v databázi, který je dán hodnotou variabilního symbolu, že platba

proběhla.

Pokud je stav příkazu 4 – Ukončeno, přiřazuji pro `payed` hodnotu -1.

Pokud je stav příkazu 0 – Nerealizováno a datum splatnosti je větší než datum splatnosti, přiřazuji pro `payed` hodnotu 0. Pomocí cron scriptu se Joomla! pravidelně provádí dotaz na tabulku `#__whmcpay_payments`, kterým zjišťuji veškeré hodnoty ze sloupce `var_symbol`, které splňují podmínku, že `payed` je rovno nule. Opět z Joomla! volám metodou PHP URL banky s aktuálními transakcemi a to pro všechny zjištěné VS.

Pokud je stav příkazu 0 – Nerealizováno a datum splatnosti je menší než datum splatnosti, přiřazuji pro `payed` hodnotu -1.

Pro administrátora jsem vytvořil pro správu plateb úlohy pro export adres klientů a faktur do POHODY a pro synchronizaci s WHMCS.

Při synchronizaci dochází k načítání chybějících faktur a plateb z WHMCS z tabulky `tblinvoices`, jedná se v podstatě o kontrolní nástroj. V případě načtení údajů do tabulky `#__whmcpay_payments` se načítá id z `tblinvoices` do `var_sym` jako VS, pokud v `tblinvoices` je `invoicenum` větší než nula ukládám do položek `final` a `payed` hodnotu 1, jinak hodnotu 0, do `proforma` se ukládá vždy hodnota 1.

Zápis hodnot pro administrátora přes toto rozhraní není povolen.

Číslo	<input type="checkbox"/>	Gateway	Date paid	Amount	Variable symbol	Payed	Proforma	Facture
1	<input type="checkbox"/>	rb	02.02.2010 16:30	1500	100039	✓	✓	✓
2	<input type="checkbox"/>	rb	04.03.2010 15:21	1500	100040	✗	✓	✗
3	<input type="checkbox"/>	rb	02.03.2010 23:52	1500	100041	✓	✓	✓
4	<input type="checkbox"/>	banktransfer	02.03.2010 22:41	8362.2	100042	✓	✓	✓
5	<input type="checkbox"/>	banktransfer	04.03.2010 15:10	3000	100043	✓	✓	✓
6	<input type="checkbox"/>	banktransfer	04.03.2010 16:50	1500	100044	✓	✓	✓
7	<input type="checkbox"/>	banktransfer	04.03.2010 16:51	1500	100045	✓	✓	✓
8	<input type="checkbox"/>	rb	04.03.2010 16:55	1500	100046	✓	✓	✓
9	<input type="checkbox"/>	banktransfer	30.11.1999 00:00	1500	100047	✗	✓	✗

Počet zobrazených: 20

Joomla! je svobodný software šířený pod GNU/GPL licenci.

Obr. 1: Přehled plateb v modulu pro administrátora

3.3 Propojení CMS Joomla! s ekonomickým a informačním systémem POHODA 2010

Dodavatelem ekonomického a informačního systému POHODA 2010 je česká softwarová společnost Stormware s.r.o.. Je určen pro malé a střední firmy. Certifikovaný je pro Windows Vista.

Dalším zadaným úkolem bylo načítat do tohoto systému faktury a v případě, že daný kontakt (údaje o klientovi) v POHODě neexistuje, naimportovat jej a tak postupně vytvořit evidenci klientů.

Pro import dat obsahuje POHODA předdefinované XML rozhraní, které je k dispozici na jejich stránkách. Vytvořené XML streamy je možné validovat pomocí volně stažitelného XML validátoru.

Na straně Joomla! nechávám generovat XML dokument buď v administraci v komponentě `whmcpay` v úloze pro export. Uložený soubor z exportu se v uživatelském prostředí POHODY naimportuje.

Pro zvýšení uživatelského komfortu jsem vytvořil utilitu. Utilita se spouští v MS WINDOWS, kde je nainstalována POHODA. V případě, že uživatel použije tlačítko Import v uživatelském rozhraní utility (Obr. 2), spustí se sekvence aktivit:

- pomocí speciální URL veřejné části Joomla! provádím: select hodnot pro XML streamy z databáze WHMCS a Joomla!
- update tabulky komponenty Joomla! whmcsPAY tabulky whmcsPAY_payments položek final a proforma
- tvorbu XML streamů, jeden pro faktury a jeden pro kontakty, a jejich uložení do adresářových struktur MS WINDOWS, konkrétně do adresáře C:\WINDOWS\temp
- volání POHODY se speciálními parametry pro import, zvlášť pro import faktur a zvlášť pro import kontaktů
- zobrazení hlášení o ukončení uživateli

POHODA vytváří log o importu jednotlivých položek. Výsledek importu lze kontrolovat v tomto logu nebo podle očekávaných hodnot přímo v uživatelské části POHODY.

V systému Joomla! v tabulce # __whmcsPAY_payments uživatel přes uživatelské rozhraní (Obr. 1) vidí stav importu faktur do POHODY. Příznak o provedení importu blokuje opakovaný import.

Při prvním spuštění utility uživatel zadává hodnoty, které jsou patrné z obrázku (Obr. 2). Při dalším spuštění jsou již údaje předvyplněny. Uložení hesla se z důvodu bezpečnosti neprovádí.

Joomla! site URL		http://whmcs.jd
Pohoda path		C:\Program Files\STORMWARE\Por...
Pohoda database name		123456789_2010
POHODA username		Admin
POHODA password		
Import		Exit

Obr. 2: Utilita

3.4 Propojení systému s registrací domén u domena.cz

Následně bylo nutné zpracovat operace s doménami (registrace, obnovení a transfér). Toto jsem zpracoval jako plugin do systému WHMCS. Pro registrátory domén je k dispozici předpřipravený balíček s rozhraním WHMCS, který si zakoupivší klient může stáhnout ze stránek výrobce.

Do zdrojového kódu balíčku je nutné přidat rozhraní daného registrátora domén. V tomto případě domena.cz, kterou provozuje nejvýznamnější česká webhostingová společnost Igunum. Rozhraní je reprezentováno pro PHP knihovnou clsIpas, která vytváří a následně validuje XML streamy ještě před dotazem na server. Ve streamu se definuje uživatel domény, administrační kontakt, dále pak samotná doména, délka její registrace a hlavně samozřejmě o jakou operaci s doménou se jedná, jestli to je registrace, prodloužení nebo přesun.

Propojení systému se striktně řídí postupy dodavatele specifikovanými v dokumentaci. Z toho důvodu by popis prací na propojení představoval duplikaci dokumentace. Problémem pro mne bylo zvládnutí složité syntaxe.

3.5 Komponenta do CMS Joomla! na správu virtuálních serverů

Komponenta do systému CMS Joomla! má zajišťovat správu virtuálních serverů. Konkrétně se jedná o prvotní založení účtů pro ftp, databázi, e-mail a statistiku návštěvnosti webu a odeslání informačního e-mailu klientovi. Přičemž e-mailový účet se založí až po obdržení platby.

Její hlavní funkcí je umožnit klientovi manipulovat s jeho účty (Obr. 4). U ftp a databáze je počet účtů omezen podle zvoleného hostingového programu. Omezení se nastavují v administraci komponenty (Obr. 3). Pro e-mail a statistiku je vždy k dispozici pouze jeden účet. U e-mailu se jedná o hlavní účet pro přístup do administrace e-mailového serveru SmarterMail, kde si již klient může nastavit ostatní účty. Totéž může nastavit přímo také administrátor webu.

V současnosti jsem vytvořil a implementoval pouze light verzi z důvodů změny API rozhraní u společnosti IGNUM, která po změně údajů klientem zašle administrátorovi hostingu e-mail s požadovanou změnou. Po provedení změny administrátorem je klient informován rovněž e-mailem.

Později je plánována plně automatická práce s účty.

V databázi Joomla jsem vytvořil tabulky (Tab. 2 až 7) pro evidenci ftp, databázových, e-mailových, statistických účtů a hostingových programů. Tyto tabulky jsou propojeny s tabulkou WMCS tblhosting. Informace z tabulek se spojují pomocí #__whcp_?.hosting.id a tblhosting.id. Propojení využívám v administraci účtů k zobrazování.

Tabulka hostingových programů #__whcp_programs je propojena s tabulkou ve WHMCS tblproducts přes hodnoty #__whcp_programs.product_id a tblproducts.id. Z této tabulky se načítá pro práci v administraci položka title, která obsahuje název produktu.

Obsahy tabulek se využívají pro práci s účty ve frontendu pro klienty (Obr. 3) a v administraci účtů (Obr. 4).

Sloupec	Typ	Nulový	Výchozí	Komentáře
id	#__whcp_programs int(5)	Ne		Primární klíč
user_id	int(5)	Ne		ID uživatele
hosting_id	int(5)	Ne		ID hostingu z tabulek WHMCS
username	varchar(255)	Ne		přihlašovací jméno
checked_out	int(5)	Ne	0	Interní zamykací pole Joomla s ID uživatele editujícího položku
checked_out_time	datetime	Ne	0000-00-00 00:00:00	Čas editace uživatelem
active	int(1)	Ne	0	Pouze pro účely light verze – příznak jestli byly změny požadované klientem aplikované

Tab. 2: Základ tabulek #__whcp_?

ftp_addr	varchar(255)	Ne		Adresa ftp serveru
ftp_path	varchar(255)	Ne	/	Cesta od ftp kořene

Tab. 3: Rozšíření pro tabulku #__whcp_ftp

email_addr	varchar(255)	Ne		E-mailová adresa ve tvaru root@domena.tld
------------	--------------	----	--	---

Tab. 4: Rozšíření pro tabulku #__whcp_emails

db_name	varchar(255)	Ne		Jméno databáze
---------	--------------	----	--	----------------

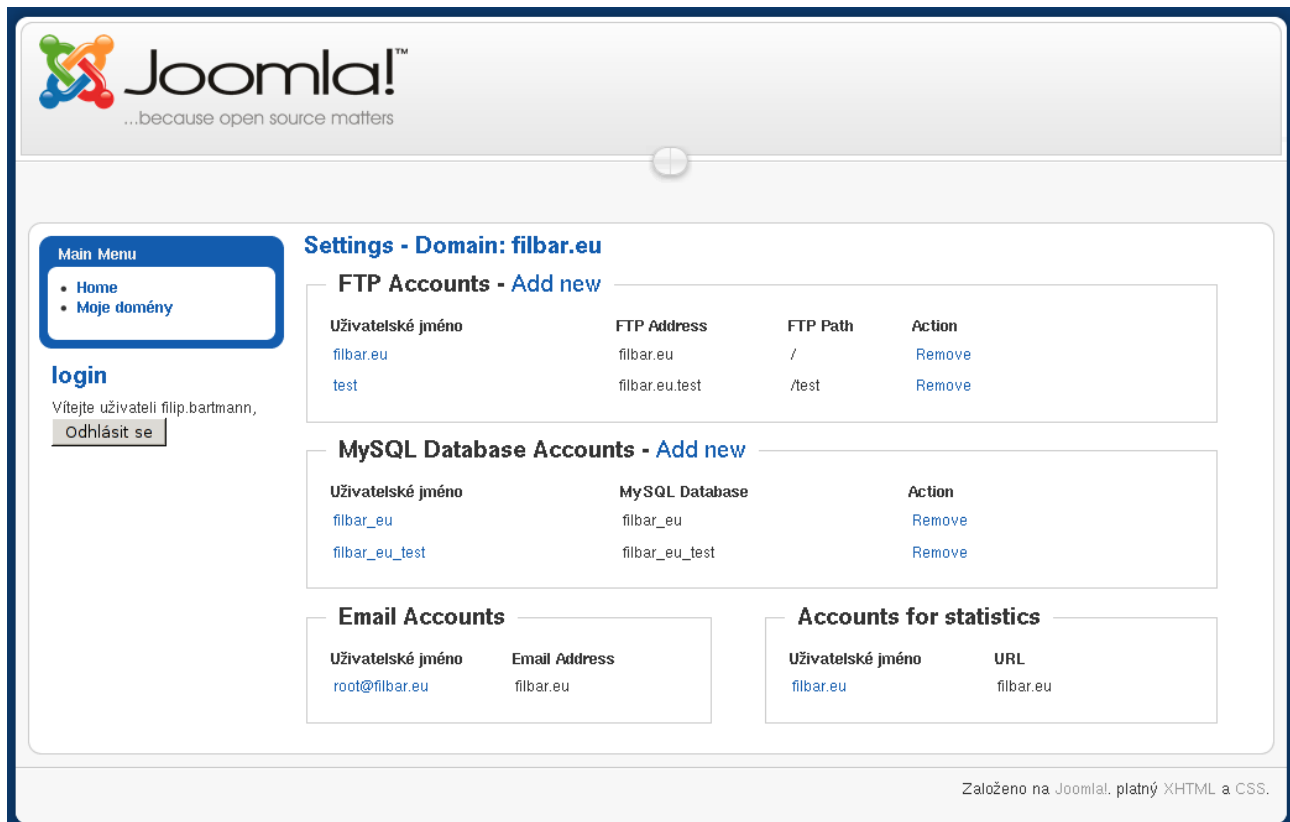
Tab. 5: Rozšíření pro tabulku #__whcp_mysqls

url	varchar(255)	Ne		Adresa statistik
-----	--------------	----	--	------------------

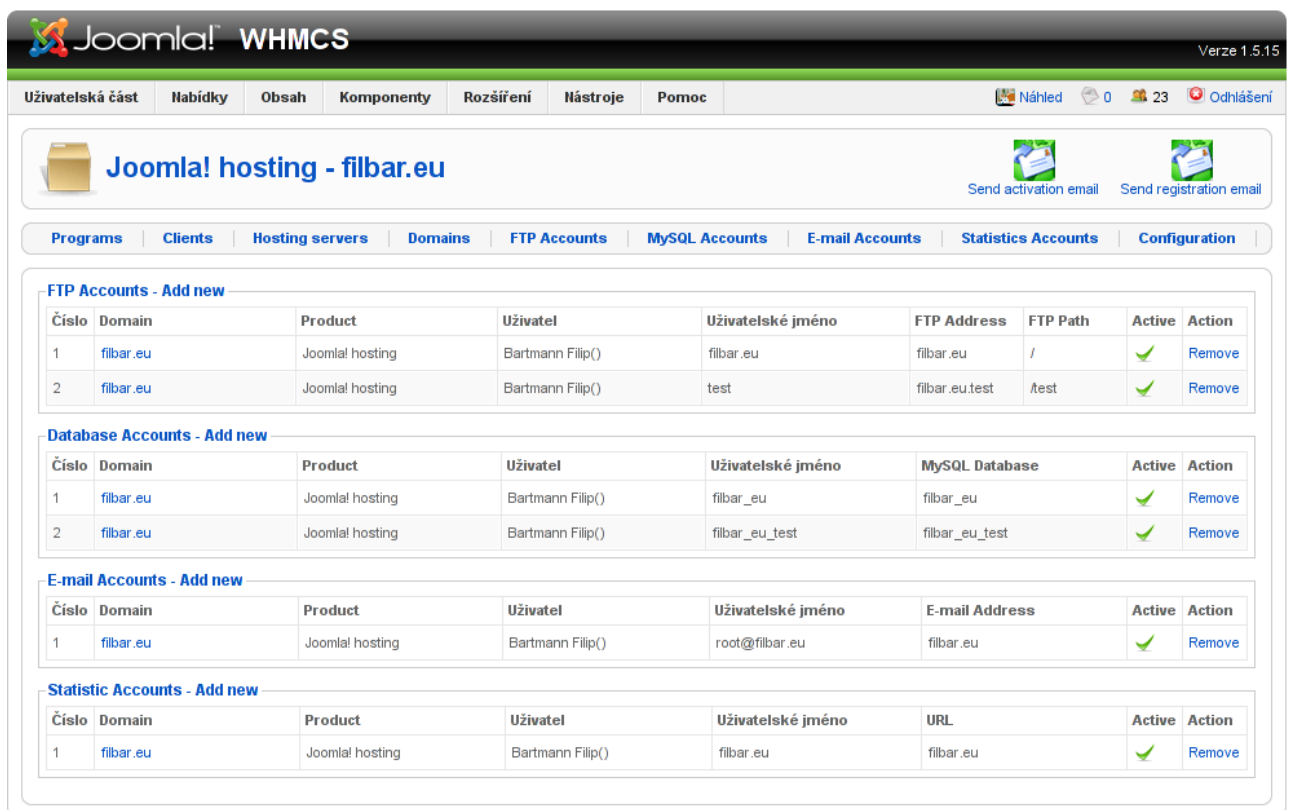
Tab. 6: Rozšíření pro tabulku #__whcp_stats

Sloupec	Typ	Nulový	Výchozí	Komentáře
id	int(5)	Ne		Primární klíč
product_id	int(10)	Ne		Id produktu z tabulky WHMCS
ftp_cnt	int(2)	Ne	1	Počet ftp účtů
mysql_cnt	int(2)	Ne	1	Počet databází
email_cnt	int(2)	Ne		Počet e-mailových účtů
checked_out	int(5)	Ne	0	Interní zamykací pole Joomla s ID uživatele editujícího položku
checked_out_time	datetime	Ne	0000-00-00 00:00:00	Čas editace uživatelem

Tab. 7: Tabulka #__whcp_programs



Obr. 3: Frontend pro správu účtů



Obr. 4: Administrace účtů

4 Znalosti ze studia.

Ze studia jsem využil hlavně znalosti programování jako takového, pak také práci s XML dokumenty a databázemi.

5 Scházející znalosti

Při práci na projektu jsem se musel hlavně naučit pracovat s dokumentací propojovaných systémů. CMS Joomla! jsem však znal již delší dobu (více než 3 roky) a programovací jazyk PHP jsem se také naučil samostudiem.

6 Dosažené výsledky

Výsledkem mé práce je systém pro webhosting SmartHost.cz provozovaný společností Testudo. Momentálně je v provozu testovací vývojová verze, která bude dle časových dispozic zadavatele uvedena do ostrého provozu.

U komponenty pro správu hostingových účtů bude z důvodu očekávané změny API provozovatele hostingu serverů nasazena pouze light verze. Až bude finální verze API, bude tato komponenta přepracována na plně automatickou správu webhostingových účtů.

Na základě probíhajícího vývoje a výstupů z testovacího provozu se zadavatel rozhodl pro převedení části funkčnosti vytvořené komponenty Joomla! whmcspay do pluginu checbank, který bude podle jeho požadavků nově vytvořen.

Po dokončení finální verze komponenty pro správu hostingových účtů bude, z důvodů provedené automatizace značně zjednodušena správa webhostingu, jak pro administrátora, tak pro klienty, jelikož se veškerá správa dosud prováděla ručně.

Mezi další přínosy patří jednotné objednávání služeb a produktů s okamžitou a periodickou fakturací z čehož vyplývá úspora času, snížení množství chyb. To umožní rozvinutí služeb, zvýšení počtu zákazníků, zlepšení konkurenceschopnosti a podpory pro zákazníky.

Na mé straně jsem si především prakticky ověřil, jak se spojují existující komponenty do jednoho celku, protože doposud jsem pracoval hlavně s jedním systémem CMS Joomla! a prohloubil jsem své programovací schopnosti.

Použitá literatura

- [1] *WHMCS-Documentation* [online]. GB: WHMCS [použito 2009-12-22]. . Dostupné z <http://wiki.whmcs.com/Main_Page>. Dokumentace k WHMCS.
- [2] *Knowledgebase-Manual Installation* [online]. GOHigher.Dostupné z : http://client.gohigheris.com/knowledgebase.php_m=knowledgebase&a=viewarticle&kbarticleid=4&nav=0,1,3
- [3] *Mechanismus XML komunikace s ekonomickým systémem POHODA* [online]. Jihlava (ČR): Stormware s.r.o. [použito 2010-02-22]. Dostupné z: <http://www.stormware.cz/xml/xmlzpracovani.aspx>
- [4] *Seznam dokladů pro Import* [online]. Jihlava (ČR): Stormware s.r.o. [použito 2010-03-22] Dostupné z: <http://www.stormware.cz/xml/dokladyimport.aspx>. Seznam dokladů pro import pro ekonomický a informační systém POHODA.
- [5] *XML Schema Validator* [online]. Verze 2.1 pro Windows, Jihlava (ČR): Stormware s.r.o. [použito 2010-04-02]. Dostupné z: <http://www.stormware.cz/xml/xmlvalidator.aspx>. Aplikace sloužící pro obecnou validaci souborů XML dle zadaného schématu.
- [6] *IPAS Server – komunikační rozhraní* [online]. Praha (ČR): IGNUMs.r.o., 2007-12-11 [použito 2010-04-23]. Dostupné z: <<http://ftp.domena.cz/doc/latest/dokumentace.pdf>>. Dokumentace.